

Claudio De Sio Cesari

Il nuovo Java Introduzione a Java (NJ–001)

<https://www.nuovojava.it>

Introduzione a Java

- Conoscere alcuni concetti informatici di base (unità 1.1).
- Saper definire il linguaggio di programmazione Java e le sue caratteristiche (unità 1.2).
- Scegliere un ambiente di sviluppo (unità 1.3).
- Saper digitare, compilare e mandare in esecuzione una semplice applicazione (unità 1.3).
- Creare semplici programmi interattivi (unità 1.4, 15).
- Interagire con l'ambiente di sviluppo ufficiale: il Java Development Kit (unità 1.5).

Hardware

- Parte fisica del computer
- Due categorie e tre tipologie
 - Input (mouse, tastiera, etc.)
 - Output (monitor, stampante, etc.)
 - Tipologia: «dentro la scatola»
(processore, memoria, scheda madre, etc)

Hardware «in the box»

- Central Processing Unit (CPU)
- Memoria
 - Principale: Random Access Memory (RAM)
 - Ausiliare (o secondaria): hard disk, SSD, etc.

Software

- Sistema operativo
- Istruzioni
- Programmi
 - Browser
 - Lettore multimediale
 - Editor
 - Videoscrittura
 - IDE

Linguaggi di programmazione

- È un mezzo che permette di comunicare delle istruzioni al computer in modo che esso le esegua nel contesto di un programma
- Vocabolario e regole (grammatica)
- Assembly, Fortran, Basic, COBOL, Pascal, C, C++, LISP, Prolog, SmallTalk, Visual Basic, Python, JavaScript, Java, C#, Kotlin

Linguaggio macchina

- Vocabolario: 0 e 1
- Sistema numerico binario
- Parole
- Diversi linguaggi macchina con grammatiche diverse
- Esigenza di un traduttore

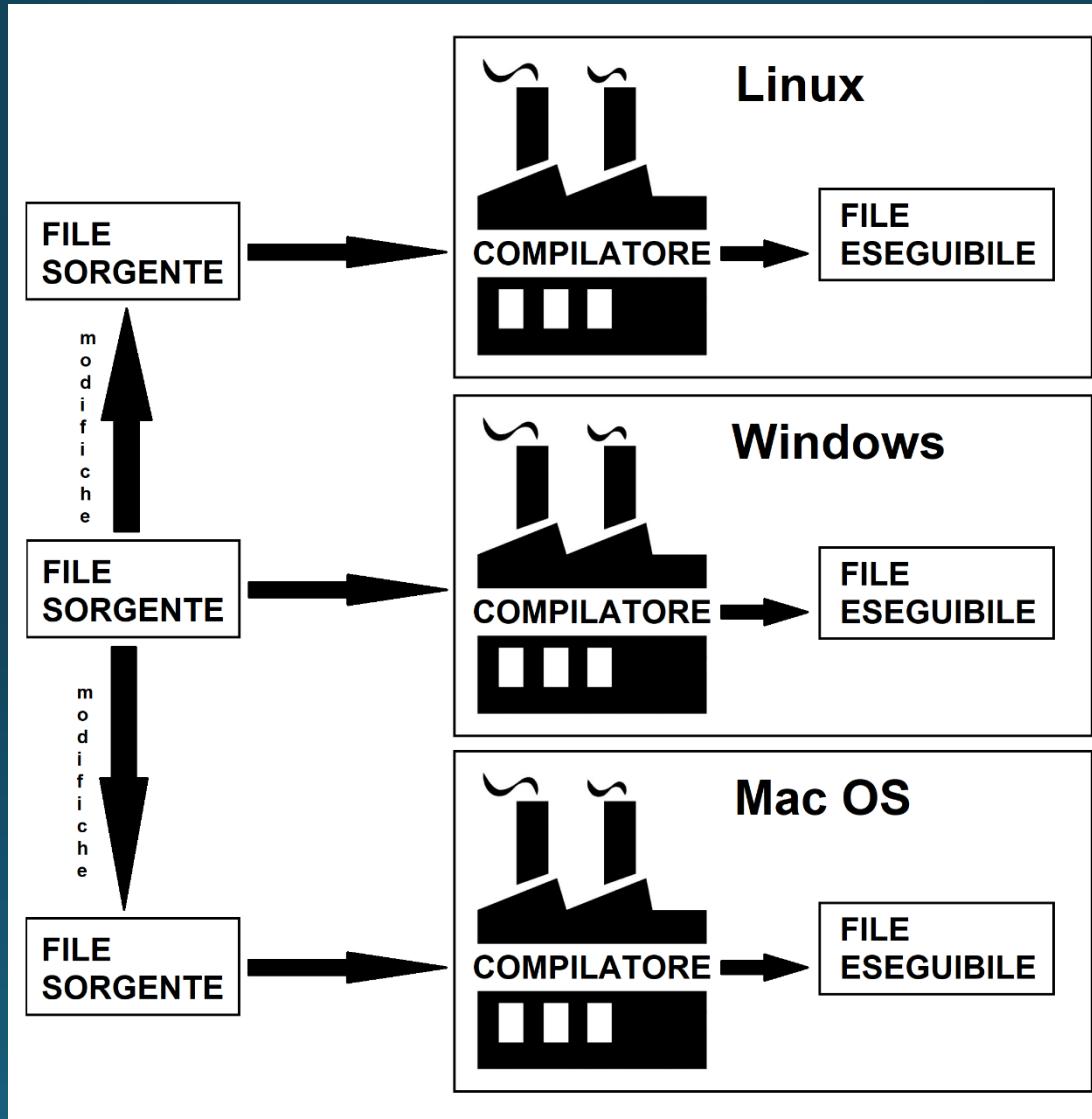
Categorizzazione dei linguaggi

- Alto livello
- Basso livello
- Linguaggi compilati
- Linguaggi interpretati
- Linguaggi compilati ed interpretati

Linguaggi compilati

- Compilatore
- Source code - Codice sorgente
- Source file - File sorgente
- Binary code - codice binario
- Binary file -File binario - eseguibile – oggetto (bin, exe, etc.)

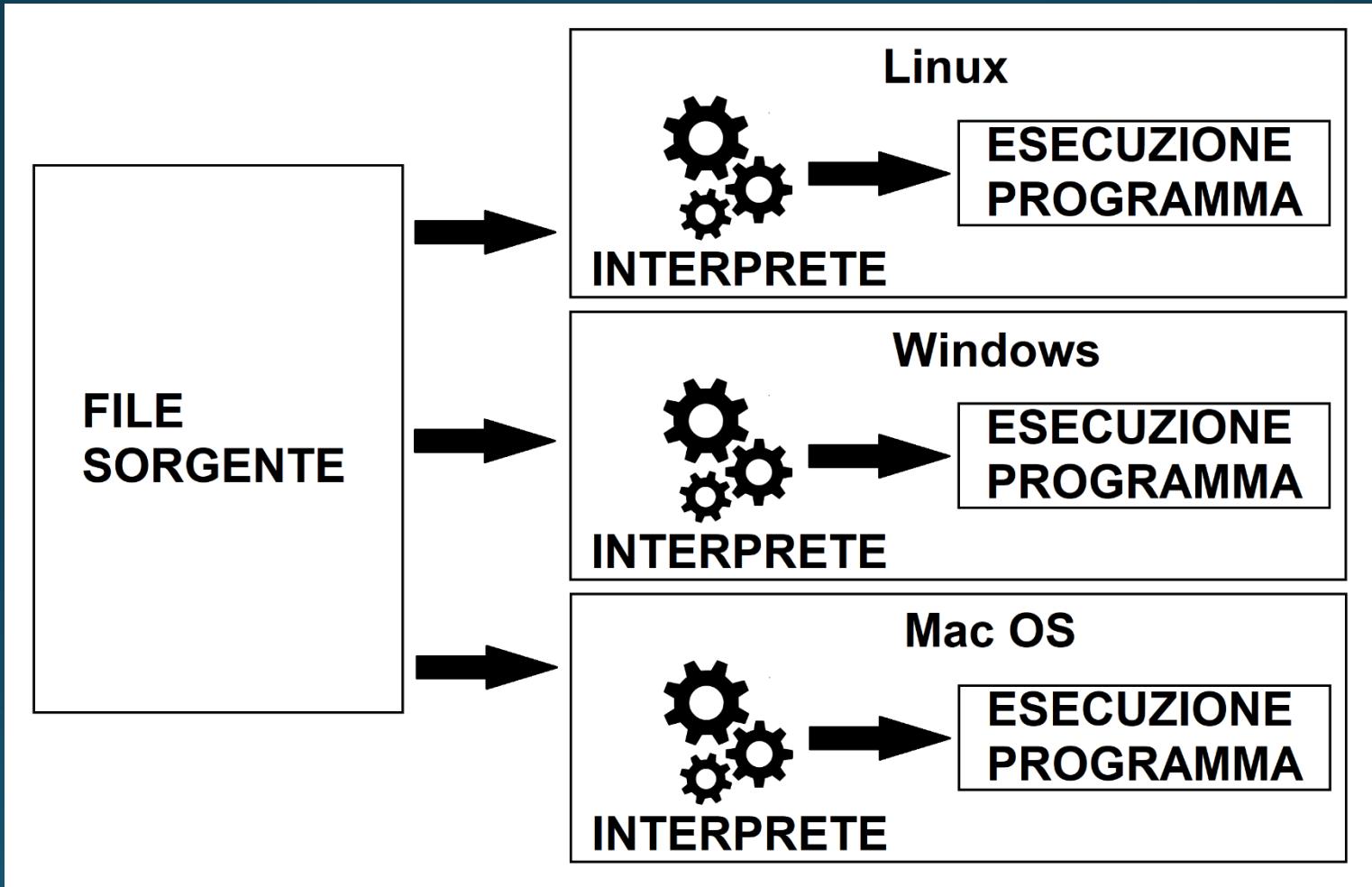
Linguaggi compilati (processo)



Linguaggi interpretati

- Interprete
 - Sviluppo più veloce
 - Esecuzione più lenta
 - Portabilità
 - L'interprete deve esistere sulla piattaforma di destinazione

Linguaggi interpretati



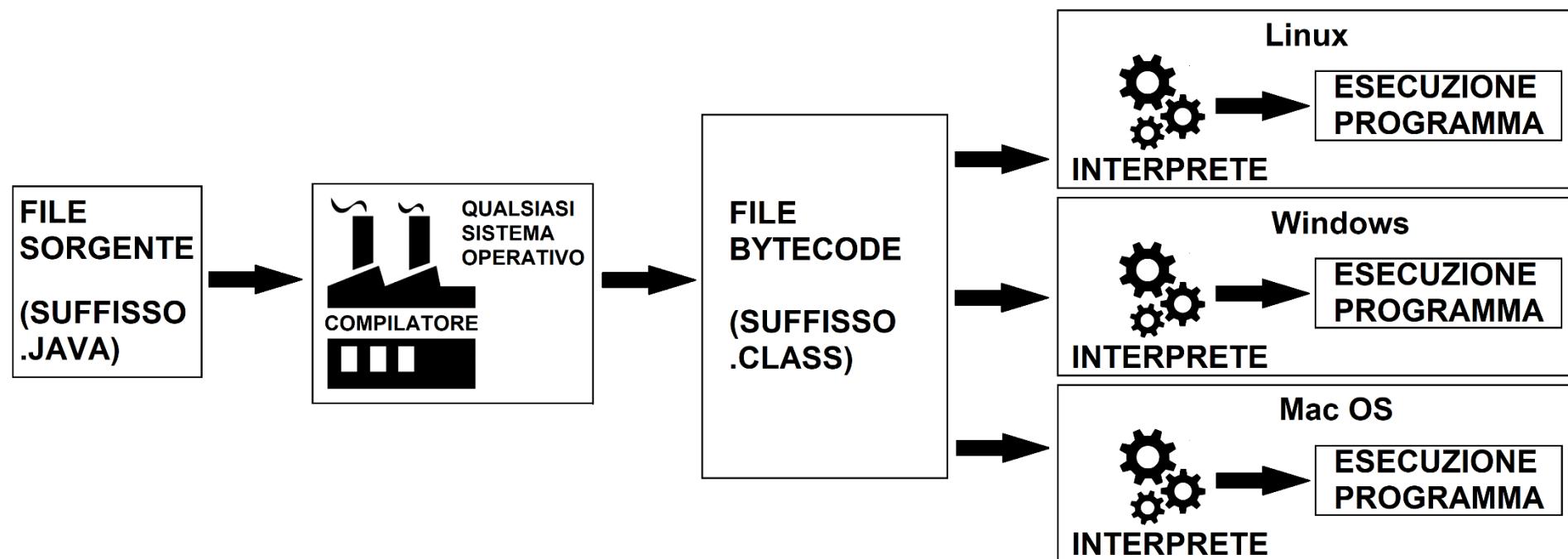
Introduzione a Java

- Con il termine “Java”, solitamente ci si riferisce:
 - al linguaggio di programmazione più usato del pianeta;
 - ad una tecnologia che include diverse “sotto-tecnologie” che si sono affermate in diversi ambiti di utilizzo del software. Oggigiorno la tecnologia Java è la più utilizzata su applicazioni aziendali (enterprise), e in tutto il mondo esistono miliardi di congegni elettronici che utilizzano tecnologia Java: smartphone, SCADA, satelliti, decoder, smart card, robot che passeggianno su Marte, etc.
 - Storia nell'appendice A de «Il nuovo Java»

Caratteristiche di Java

- Sintassi
- Robustezza
- Programmazione Object Oriented (OOP)
- Facilità di sviluppo
- Librerie e standardizzazione
- Aperto
- Indipendenza dalla piattaforma
- Java Virtual Machine (JVM)

Indipendenza dalla piattaforma e Java Virtual Machine



False convinzioni su Java

- Java è lento
- Java non è sicuro ([link 1.1](#))
- Java è prolioso
- Java è facile da imparare
- Java è a pagamento
- JavaScript è una versione semplificata di Java

Scrivere un programma

- Scrivere il codice sorgente: scriveremo il codice sorgente della nostra applicazione utilizzando un **editor**, ovvero un'applicazione che ci permette di scrivere codice Java.
- Salvare il file sorgente: tramite l'editor, salveremo il nostro file con suffisso *.java* nella memoria secondaria, presumibilmente in una cartella da noi creata sul nostro hard disk.
- Compilare il nostro file sorgente: una volta salvato il nostro file sorgente, dobbiamo utilizzare il compilatore per generare un file contenente il bytecode. Questo file avrà automaticamente un suffisso *.class*.
- Eseguire il nostro programma: potremo infine mandare in esecuzione il programma invocando l'interpretazione della Java Virtual Machine.

Ambiente di sviluppo

- Soluzione classica
 - Java Development Kit (JDK) Standard Edition
 - Un editor che non supporta stili (Blocco Note)
- Soluzione per chi inizia
 - EJE (Everyone's Java Editor)
- Soluzione per esperti
 - Netbeans
 - Eclipse
 - IntelliJ

Programmiamo con EJE

- EJE 4.0 -

File Edit Search Insert View Development Tools Help

Punto.java Esercizio10N.java Polilinea.java PolilineaNonValidaException.java Righello.java Segmento.java SegmentoNonValidoException.java CartesianPlane.java

File to run Current File

1

2

3

```
import java.awt.Dimension;
import java.util.*; import java.awt.*;

public class CartesianPlane {
    private final static int LENGTH = 600;
    private final static int MID_LENGTH = LENGTH/2;
    private final static int GAP = LENGTH/10;
    private JFrame frame;
    private JPanel panel;
    private ArrayList<Punto> points;
    private ArrayList<Segmento> segments;

    public CartesianPlane() {
        frame = new JFrame("Cartesian Plane");
        panel = new CartesianPlanePanel();
        points = new ArrayList<>();
        segments = new ArrayList<>();
        setup();
        addDetails();
    }

    private void setup() {
```

C:\Users\claud\OneDrive\Il Nuovo Java 15\Codice\capitolo_10\esercizi\10.n\CartesianPlane.java built.
Process Terminated with no errors.
Output Directory: C:\Users\claud\OneDrive\Il Nuovo Java 15\Codice\capitolo_10\esercizi\10.n

Save Ctrl-S
Build F7
Execute F9
Cut Ctrl-X
Copy Ctrl-C
Paste Ctrl-V
Delete Ctrl-H
Comment Selection Ctrl-F12
Select All Ctrl-A
To Upper Case Ctrl-U
To Lower Case Ctrl-L
Invert Upper-Lower Ctrl-J
Format Code Ctrl+Maiuso-F
Close File Ctrl-W

Programmiamo: Scrivere il codice sorgente

- Creare un nuovo file con EJE.
- Verrà visualizzato nuovo “foglio”, dove poter scrivere il nostro programma.
- Possiamo quindi copiare il codice sorgente del programma “HelloWorld” nell’area 2.
- Copiare a mano il codice riga dopo riga, dopo averne letta la relativa spiegazione. In questo modo si prenderà più coscienza di quello che scrive.

Programmiamo «Hello World»

```
public class HelloWorld
{
    public static void main(String args[])
    {
        System.out.println("Hello World!");
    }
}
```

«Hello World» versione Java

```
public class HelloWorld {  
    public static void main(String args[]) {  
        System.out.println("Hello World!");  
    }  
}
```

Programmiamo: tutti i passi

- Pre requisito abilitare visualizzazione suffisso
- Creare un cartella di lavoro
- Salvare il file nella cartella giusta
- Aprire la cartella di lavoro in EJE
- Compilare
- Visualizzare risultato
- Eseguire
- Altri ambienti di sviluppo

Secondo programma (interattivo)

```
import java.util.Scanner;

public class MoltiplicaDueInteri {
    public static void main(String args[]) {
        System.out.println("Ciao, so moltiplicare due numeri interi.");
        System.out.println("Scrivi il primo valore e batti invio");
        Scanner keyboardScanner = new Scanner(System.in);
        int input1;
        input1 = keyboardScanner.nextInt();
        System.out.println("Scrivi il secondo valore e batti invio");
        int input2 = keyboardScanner.nextInt();
        int risultato = input1 * input2;
        System.out.println("Il risultato è:");
        System.out.println(risultato);
    }
}
```

Non è mica così semplice!

- Approfondimento 1.3 «Possibili problemi in fase di compilazione ed esecuzione»
- Testare l'applicazione usando input diversi ed analizzare gli output
- Mettere in un lista tutti i problemi che si sono presentati
- Sapete risolverli?

JDK

- Installazione (appendice B o [video](#))
- Struttura del JDK
- Differenza con JRE e JVM
- Provare a compilare ed eseguire i programmi da riga di comando
- Creare la classe *SommaTreInteri* con blocco note e JDK
- Riassumere tutte le azioni intraprese per sviluppare con questo ambiente

Sommario

- Conoscere alcuni concetti informatici di base (unità 1.1).
- Saper definire il linguaggio di programmazione Java e le sue caratteristiche (unità 1.2).
- Scegliere un ambiente di sviluppo (unità 1.3).
- Saper digitare, compilare e mandare in esecuzione una semplice applicazione (unità 1.3).
- Creare semplici programmi interattivi (unità 1.4, 15).
- Interagire con l'ambiente di sviluppo ufficiale: il Java Development Kit (unità 1.5).